LOW LEVEL DOCUMENT

Introduction:

1.1. What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the

actual program code for Food Recommendation System. LLD describes the class diagrams with the

methods and relations between classes and program specs. It describes the modules so that the

programmer can directly code the program from the document.

1.2. Scope

Low-level design (LLD) is a component-level design process that follows a step-by-

step refinement process. This process can be used for designing data structures, required software

architecture, source code and ultimately, performance algorithms. Overall, the data organization

may be defined during requirement analysis and then refined during data design work

2. Architecture

ML PIPELINE

DATA TRANSFORMATION

MODEL PUSHER

MODEL TRAINER

MODEL EVALUATION

DATA INGESTION

DATA VALIDATION

DATA INGESTION:

Data ingestion is the process of collecting and importing data from various sources into a storage system, such as a data warehouse or data lake, where it can be further processed and analyzed. The data sources can be structured, semi-structured or unstructured data, and can come from a wide range of sources such as databases, APIs, files, sensors, logs, and more.

The process of data ingestion typically involves extracting data from the source, transforming it into a common format, and loading it into the destination storage system. This process can be automated using tools like ETL (extract, transform, load) or ELT (extract, load, transform) processes.

Data ingestion is a critical step in the data processing pipeline because it sets the foundation for further analysis and decision-making. By ingesting data from diverse sources, organizations can gain a comprehensive view of their operations, customers, and market trends. This data can then be used to identify patterns, make predictions, and drive business growth.

DATA VALIDATION

Data validation is the process of ensuring that data entered into a computer system is accurate, complete, and consistent with certain predefined rules and constraints. The main purpose of data validation is to improve the quality and reliability of data by identifying errors, inconsistencies, and missing information before it is processed or stored in a database.

There are various techniques and tools used for data validation, including:

1. Data type validation: This involves checking that the data entered matches the expected data type, such as checking that a phone number is a numeric value and not a string.
2. Range validation: This involves checking that the data entered falls within a specified range or set of values, such as checking that a date is within a certain range of years.
3. Format validation: This involves checking that the data entered matches a specific format or pattern, such as checking that an email address includes an '@' symbol.
4. Cross-field validation: This involves checking that data entered in one field is consistent with data entered in other fields, such as checking that the state and zip code entered match the city entered.
5. Code validation: This involves checking that the data entered matches a specific code or standard, such as checking that a product code matches a known product in a database.

Overall, data validation is an important step in ensuring the accuracy and reliability of data, and can help prevent errors and inconsistencies that could lead to problems down the line.

DATA TRANNSFORMATION

Data transformation refers to the process of converting or modifying data from one format, structure, or type to another in order to make it more useful for analysis, visualization, or other purposes.

Data transformation can include various techniques such as data cleaning, feature selection, feature engineering, data normalization, data aggregation, and data summarization. These techniques are used to transform raw data into a form that is suitable for analysis and modeling.

Data cleaning involves identifying and correcting errors, inconsistencies, and missing values in the data. Feature selection involves identifying the most important variables or features that contribute to the prediction of an outcome. Feature engineering involves creating new features or variables that can improve the accuracy of the model. Data normalization involves scaling the data so that it falls within a specific range or distribution.

Data aggregation involves grouping data into categories or clusters based on specific criteria. Data summarization involves summarizing the data into a more compact form, such as by computing the mean or median of a group of values.

Data transformation is a crucial step in the data analysis process as it helps to ensure that the data is accurate, consistent, and relevant for the analysis. It also helps to improve the accuracy and reliability of the models built using the data.

MODEL TRAINER

A model trainer is a software tool or framework used to build machine learning models. It allows users to specify the architecture of the model, the data to be used for training, and the optimization algorithm to be used to learn the parameters of the model.

The model trainer takes the input data and applies various transformations and pre-processing steps, such as normalization or feature extraction, to prepare the data for training. It then trains the model using the specified algorithm and hyperparameters. During training, the model is repeatedly presented with training data, and the algorithm adjusts the model parameters to minimize a predefined loss function, which measures the difference between the predicted output and the actual output.

The performance of the trained model is evaluated using a validation set or test set, which contains data that was not used during training. The model trainer can help users to fine-tune the model parameters and choose the best model based on the evaluation results.

Model trainers can be used for a wide range of machine learning tasks, including classification, regression, clustering, and natural language processing. Some popular model trainers include TensorFlow, PyTorch, scikit-learn, Keras, and Apache MXNet.

MODEL EVALUATION

Model evaluation is the process of measuring the performance of a machine learning model on a given dataset. The goal of model evaluation is to determine how well the model can generalize to new, unseen data, and to identify potential issues such as overfitting or underfitting.

There are several metrics used to evaluate the performance of a model, depending on the type of problem being solved. For example:

* In a classification problem, commonly used metrics include accuracy, precision, recall, F1-score, and ROC curve.
* In a regression problem, commonly used metrics include mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and R-squared.

To evaluate a model, the dataset is usually split into a training set and a test set, or a training set, a validation set, and a test set. The model is trained on the training set, and the performance is evaluated on the test set or the validation set.

MODEL PUSHER:

It allows us to push the model on aws ,azure etc.

Model deploy:

We will be deploying the model to AWS.